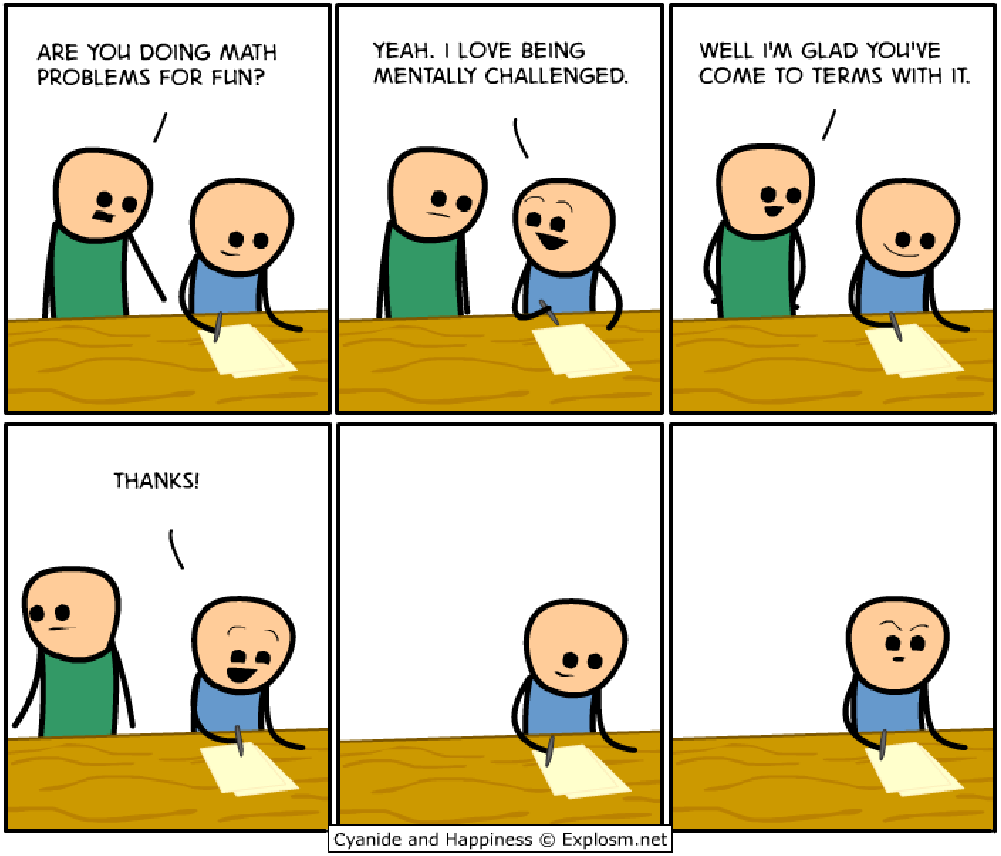
# Fall 2015

## Lecture 13: Primes, GCDs



- Compute $78365^{81453}$

$a^{i+j} \bmod m = (a^i \bmod m)(a^j \bmod m) \bmod m$

- Compute $78365^{81453} \bmod 104729$ (with $a = 78365$, $k = 81453$, $m = 104729$)

- Output is small
  - need to keep intermediate results small

81453 times:

$$\begin{aligned} n_1 &= a \bmod m \\ n_2 &= a \cdot n_1 \bmod m = a^2 \bmod m \\ n_3 &= a \cdot n_2 \bmod m = a^3 \bmod m \\ &\vdots \\ n_k &= a \cdot n_{k-1} \bmod m = a^k \bmod m. \end{aligned}$$

# repeated squaring – small and fast

$a \equiv b \bmod$ iff $a \bmod m = b \bmod m$

$(a \bmod m)^2 \equiv a^2 \bmod m$

$a^2 \bmod m = (a \bmod m)^2 \bmod m$

Since a mod m ≡ a (mod m) for any a

we have $a^2 \bmod m = (a \bmod m)^2 \bmod m$

and $a^4 \bmod m = (a^2 \bmod m)^2 \bmod m$

and $a^8 \bmod m = (a^4 \bmod m)^2 \bmod m$

and $a^{16} \bmod m = (a^8 \bmod m)^2 \bmod m$

and $a^{32} \bmod m = (a^{16} \bmod m)^2 \bmod m$

Can compute $a^k \bmod m$ for $k = 2^i$ in only $i$ steps

$k = 2^i + 2^j$

$a^{2^i}$ $a^{2^j}$

$k = 1001011$

$a^{2^6}$ $a^{2^3}$ $a^{2^1}$ $a^{2^0}$

# fast exponentiation algorithm

ModPow(a, k, m) should compute $a^k \bmod m$.

If $k == 0$ then

return 1

If $(k \bmod 2 == 0)$ then

return ModPow($a^2 \bmod m, k/2, m$) → $(a^2 \bmod m)^{k/2} \bmod m = a^k \bmod m$ (k/2 → integer)

else

return $(a \times$ ModPow$(a, k-1, m)) \bmod m$

$(a \cdot (a^{k-1} \bmod m)) \bmod m$

ModPow$(a, k-1, m)$ → $a^{k-1} \bmod m$

call ModP(a, 81452, m)
ModP($a^2$, 40726, m)

$$k = 81453$$
$$= (10011111000101101)_2$$
$$= 2^{16} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^{9} + 2^{5} + 2^{3} + 2^{2} + 2^{0}$$

Total # of arithmetic operations $\sim 4 \times 16 = 64$

## Another way:

$$81453 = 2^{16} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^{9} + 2^{5} + 2^{3} + 2^{2} + 2^{0}$$

$$a^{81453} = a^{2^{16}} \cdot a^{2^{13}} \cdot a^{2^{12}} \cdot a^{2^{11}} \cdot a^{2^{10}} \cdot a^{2^{9}} \cdot a^{2^{5}} \cdot a^{2^{3}} \cdot a^{2^{2}} \cdot a^{2^{0}}$$

$a^{81453}$ mod m=

(…(((((a$^{2^{16}}$ mod m ·

a$^{2^{13}}$ mod m ) mod m ·

a$^{2^{12}}$ mod m) mod m ·

a$^{2^{11}}$ mod m) mod m ·

a$^{2^{10}}$ mod m) mod m ·

a$^{2^{9}}$ mod m) mod m ·

a$^{2^{5}}$ mod m) mod m ·

a$^{2^{3}}$ mod m) mod m ·

a$^{2^{2}}$ mod m) mod m ·

a$^{2^{0}}$ mod m) mod m

The fast exponentiation algorithm computes $a^n \bmod m$ using $O(\log n)$ multiplications $\bmod m$

# primality

An integer *p* greater than 1 is called *prime* if the only positive factors of *p* are 1 and *p*.

$p = 13$ prime

$p = 15$ not prime $3 \mid 15$

A positive integer that is greater than 1 and is not prime is called *composite*.

26 composite $13 \mid 26$

# Fundamental Theorem of Arithmetic

Every positive integer greater than 1 has a unique prime factorization

| | | |
|---|---|---|
| 48 | = | $2 \cdot 2 \cdot 2 \cdot 2 \cdot 3$ |
| 591 | = | $3 \cdot 197$ |
| 45,523 | = | 45,523 |
| 321,950 | = | $2 \cdot 5 \cdot 5 \cdot 47 \cdot 137$ |
| 1,234,567,890 | = | $2 \cdot 3 \cdot 3 \cdot 5 \cdot 3{,}607 \cdot 3{,}803$ |

If $n$ is composite, it has a factor of size at most $\sqrt{n}$.

$n = p_1 p_2 \cdots p_k$

Since $n$ is comp $k \geq 2$

if $p_1, p_2 > \sqrt{n}$ then $n \geq p_1 p_2 > n$

Contradiction.

## There are an infinite number of primes.

Proof by contradiction:

Suppose that there are only a finite number of primes: $p_1, p_2, \dots, p_n$

$$p_1 p_2 \cdots p_n + 1 = \underbrace{p_{i_1} p_{i_2} \cdots p_{i_k}}_{\text{prime factori}}$$

$$\underbrace{(p_1 \cdots p_n + 1)}_{\text{mod } p_{i_1} = 0} \text{ mod } p_{i_1} = \underbrace{p_{i_1} \cdots p_{i_k} \text{ mod } p_{i_1}}_{0}$$

$1 = 0$

contradiction.

# Famous Algorithmic Problems

- Primality Testing
  - Given an integer $n$, determine if $n$ is prime
  - Fermat's little theorem test:

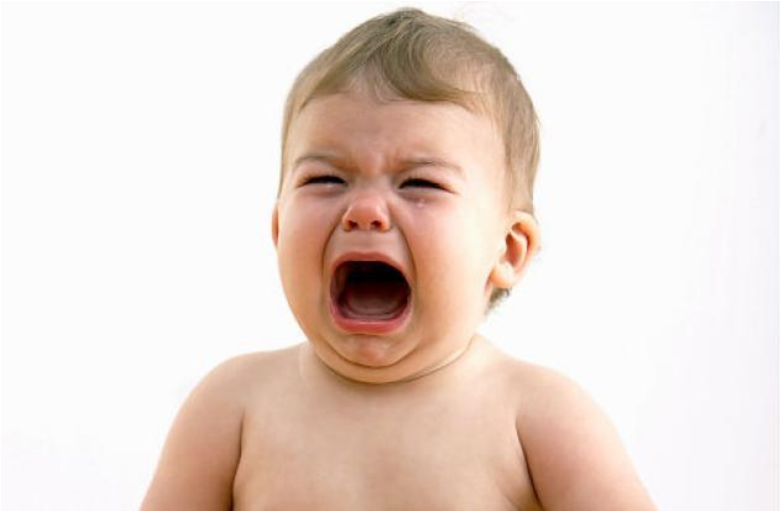
    If $p$ is prime and $a \neq 0$, then $a^{p-1} \equiv 1 \pmod{p}$

- Factoring
  - Given an integer $n$, determine the prime factorization of $n$

# Factoring

## Factor the following 232 digit number [RSA768]:

1230186684530117755130494958384962720772
8535695953347921973224521517264005072636
5751874520219978646938995647494277406384
5925192557326303453731548268507917026122
1429134616704292143116022212404792747377
94080665351419597459856902143413



1230186684530117755130494958384962720772853569595334792197322452151726400507263657518745202199786469389956474942774063845925192557326303453731548268507917026122142913461670429214311602221240479274737794080665351419597459856902143413

=

3347807169895689786044169848212690817704794983713768568912431388982883793878 … 17
4308773781446799489

×

36746043666799590428244633799 … 643
43087642676032283815739666511 … 968
102700927987363089172

GCD(a, b):

Largest integer $d$ such that $d \mid a$ and $d \mid b$

- GCD(100, 125) = 25
- GCD(17, 49) = 1
- GCD(11, 66) = 11
- GCD(13, 0) = 13
- GCD(180, 252) = 36

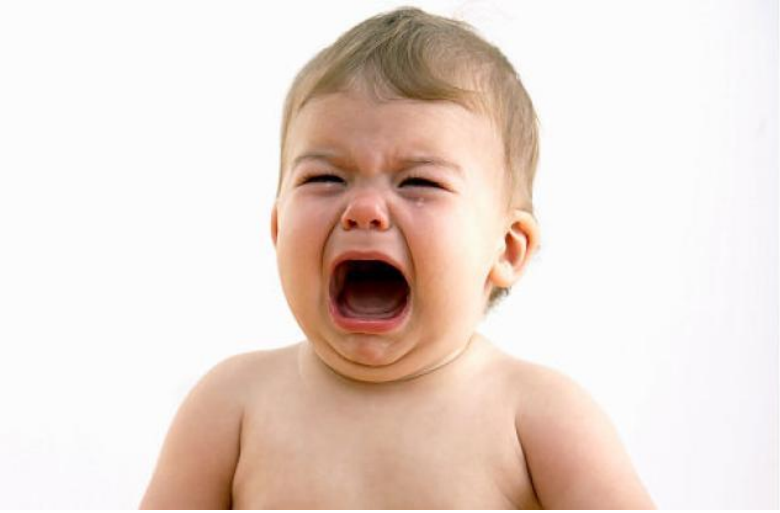
$2^2 \cdot 3^2 \cdot 5$ $\quad$ $2^2 3^2 7$

# GCD and Factoring

$a = 2^3 \cdot 3 \cdot 5^2 \cdot 7 \cdot 11 = 46{,}200$

$b = 2 \cdot 3^2 \cdot 5^3 \cdot 7 \cdot 13 = 204{,}750$

$GCD(a, b) = 2^{\min(3,1)} \cdot 3^{\min(1,2)} \cdot 5^{\min(2,3)} \cdot 7^{\min(1,1)} \cdot 11^{\min(1,0)} \cdot 13^{\min(0,1)}$



Factoring is expensive!

Can we compute GCD(a,b) without factoring?

# Useful GCD Fact

If $a$ and $b$ are positive integers, then

$$\gcd(a, b) = \gcd(b, a \bmod b)$$

**Proof:**

By definition $a = (a \text{ div } b) \cdot b + (a \bmod b)$

If $d \mid a$ and $d \mid b$ then $d \mid (a \bmod b)$.

If $d \mid b$ and $d \mid (a \bmod b)$ then $d \mid a$.

# Euclid's Algorithm

Repeatedly use the GCD fact to reduce numbers until you get $\mathrm{GCD}(x, 0) = x$.

$a > b$

$\mathrm{GCD}(a,b) = \mathrm{GCD}(b, a \bmod b)$

GCD(660,126) $= \mathrm{GCD}(126, 660 \bmod 126)$

$= \mathrm{GCD}(126, 30)$

$= \mathrm{GCD}(30, 6)$

$= \mathrm{GCD}(6, 0) = 6$

# Euclid's Algorithm

GCD(x, y) = GCD(y, x mod y)

```
int GCD(int a, int b){ /* a >= b, b > 0 */
    int tmp;
    while (b > 0) {
        tmp = a % b;
        a = b;
        b = tmp;
    }
    return a;
}
```

Example: GCD(660, 126)

If *a* and *b* are positive integers, then there exist integers ***s*** and ***t*** such that

$$\gcd(a,b) = \mathbf{s}a + \mathbf{t}b$$

# Extended Euclidean Algorithm

- Can use Euclid's Algorithm to find $s, t$ such that $\gcd(a, b) = sa + tb$
- e.g. gcd(35,27):

| | |
|---|---|
| 35 = 1 • 27 + 8 | 35 - 1 • 27 = 8 |
| 27= 3 • 8 + 3 | 27- 3 • 8 = 3 |
| 8 = 2 • 3 + 2 | 8 - 2 • 3 = 2 |
| 3 = 1 • 2 + 1 | 3 - 1 • 2 = 1 |
| 2 = 2 • 1 + 0 | |

- Substitute back from the bottom

1 = 3 - 1 • 2 = 3 – 1 (8 - 2 • 3) = (-1) • 8 + 3 • 3
= (-1) • 8 + 3 (27- 3 • 8 ) = 3 • 27 + (-10) • 8
= 3 • 27 + (-10) • (35 - 1 • 27) = -10 • 35 + 13 • 27

# Multiplicative Inverse $\mathrm{mod}\ m$

Suppose $\mathrm{GCD}(a, m) = 1$

By Bézout's Theorem, there exist integers $s$ and $t$ such that $sa + tm = 1$.

$s \bmod m$ is the multiplicative inverse of $a$:

$$1 = (sa + tm) \bmod m = sa \bmod m$$

Solving $ax \equiv b \pmod{m}$ for unknown $x$ when $\gcd(a, m) = 1$.

1. Find $s$ such that $sa + tm = 1$
2. Compute $a^{-1} = s \bmod m$, the multiplicative inverse of $a$ modulo $m$
3. Set $x = (a^{-1} \cdot b) \bmod m$

# Example

Solve: $7x \equiv 1 \pmod{26}$